OGIP Calibration Memo CAL/GEN/2003-001

Automated Delivery of Calibration Data to the CALDB

edited by:

Lorella Angelini & Mike Corcoran

HEASARC
Codes 662
NASA/GSFC,
Greenbelt, MD20771

Last Update: 2003 Sep 23

SUMMARY

This documents how to deliver CALDB data files into the CALDB staging area for automatic release to the public CALDB
Intended audience: General users, Data producers, OGIP programmers and authors of data analysis s/w.

LOG OF SIGNIFICANT CHANGES

Release Date	**Sections Changed**	**Brief Notes**
2003 Sep 10	All	First version
2003 Sep 10	All	some clarifications

Contents

1 Overview

This document provides a guide for delivering data to the CALDB staging area for automated release into the public CALDB. This automated mechanism was first used by the SWIFT mission starting in 2003. Any project/mission which wishes to use this automated release should identify a responsible party (the "Project Calibration Team") to provide calibration files into an appropriate directory in the CALDB staging area. This directory will have to be set up by the CALDB manager who can be identified from the CALDB home page, `http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_intro.html`.
The CALDB staging directory structure has to mirror the directory structure of the mission's CALDB data directories.

Each delivery consists in the following :

- one or more Calibration data files copied to the CALDB staging area in the appropriate subdirectory
- a Calibration Index File
- An email sent to the `caldbingest@olegacy.gsfc.nasa.gov` address.

This document describes what the project should do to deliver one or more calibration files for one or more particular instruments, and how to signal to the CALDB manager that the delivery is available in the staging area for automated release to the public CALDB.

2 Calibration Data File Requirements

Unless the data files are "Primary Calibration Files" (PCF) which are to be stored for historical purposes (but which are not meant to be used by analysis software), the data files are expected to be stored in subdirectories in the "Basic Calibration File" (BCF) or the "Calibration Products File" (CPF) directories (see `http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/docs/summary/cal_gen_92_003_summary.html`
for CPF/BCF information and guidelines).

For BCF or CPF data, these data

1. must be in FITS format and include all required keywords (see `http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_keywords.html`
   for a list of required keywords);
2. the data files must be verified as valid FITS files (using a routine like the fverify tool which is part of the HEASOFT package)

3. must contain valid calibration data;
4. the FITS data must contain valid CHECKSUM and DATASUM keywords in each FITS header;
5. each file should be associated with only 1 CALDB subdirectory.

The above 5 steps are the responsibility of the Project Calibration Team.

3 Delivery to the CALDB Staging Area

The CALDB staging area for a given mission and for a given instrument is organized following standard CALDB usage as follows: `$CALDB/data/<mission>/<instrument>/`, (the "instrument directory") where `<mission>` is the string which has been adopted to name the mission, and `<instrument>` is a string which has been adopted to describe a given element associated with this mission. Each mission will have one or more instruments and an instrument can include common elements such as the entire spacecraft (`<instrument>=spacecraft`) or an X-ray telescope (`<instrument>=xrt`) which may be shared by two or more sub-components. Calibration data releases need to be on an instrument-by-instrument basis.

After verifying a calibration file (or calibration files) for release to the CALDB, the Project Calibration Team data copies each data file to the appropriate subdirectory in the CALDB staging area. For each `<instrument>` for which new calibration data are being staged, the project calibration team needs to supply a valid Calibration Index File (CIF). This CIF needs to include not only the new calibration data being staged, but also needs to include, *at mininum*, all valid calibration data for that `<instrument>` and for that `<misssion>`. It is strongly encouraged that all calibration data (valid an invalid) which may have been released for that particular `<instrument>` and for that `<misssion>` be included in the CIF (with the invalid or superceded data marked appropriately using the `CAL_QUAL` keyword).

The delivery should include two identical caldb index files named as follows:

- caldb.indx. This file is used by the CALDB software to query the CALDB and access/retrieve data from the CALDB.
- `caldb.indxYYYYMMDD` where `YYYYMMDD` is the delivery date. In the public CALDB this file will be placed in a subdirectory named `index` within the `instrument` directory. All the previous deliveries of `caldb.indexYYYYMMDD` files are kept in the `index` subdirectory to allow to recovery of information on what files were present in CALDB and what was their quality setting in each release. In this way files produced with a specific delivery of CALDB can be reproduced by selecting the specific caldb.indexYYYYMMDD.

4 Email Notification of the CALDB Manager

After data and index files have been copied to the CALDB staging area an email should be sent to `caldbingest@legacy.gsfc.nasa.gov` The subject of the e-mail should contain mission name and date of the delivery (YYYYMMDD) preceeded by the string "caldb update". For example :

```
Subject: caldb update ASCA19931220
```

indicates that the ASCA calibration data file were staged on December 20 1993.

The body of the email should a keyword `instrument` which gives the CALDB identifier of the instrument for a given mission and, on the next line, a keyword `caldbindexfile` which gives the name of the CIF appropriate for the particular data delivery. Following the `caldbindexfile` keyword (on a new line) is the `newfile` tag. After this tag, that the name and paths of each data file are given, one file per line. The `endnewfile` tag signifies that all data files for this particular `instrument` for this particular release have been given. A release can include one or more datafiles for one or more instruments.

For example the email notification for a release of ASCA data for the SIS and GIS instruments might look like the following:

```
From: ASCA calibration team
To: caldbingest@olegacy.gsfc.nasa.gov
Subject:  caldb update ASCA19931220

   instrument= GIS
   caldbindexfile= caldb.indx931220
   newfile
   gis/cpf/gis_center256.rmf
   gis/cpf/93dec20/g2_1p.arf
   endnewfile
   instrument= SIS
   caldbindexfile= caldb.indx931220
   newfile
   sis/cpf/s0c3g0p20e0_512_1c.rsp
   sis/bcf/sis0c3p40_290296.fits
   sis/bcf/bgd/s0bgd_14i.evt
   endnewfile
```

5 Delivery

The first delivery should contain all the CALDB files. The updates either because a new version of the file has been released or because a new file has been added should contain ONLY the new data file. The caldb index appropariate for that instrument should reflect what is changed in caldb :

- New file. The caldb index file should include entries which document the new file
- File Update. The caldb index file should include new entries documenting the updated file. The `CAL_QUAL` value for the old file should be set to a higher value (usually changed from `CAL_QUAL=0` to `CAL_QUAL=5`.)

6 Release to the Public CALDB

After the data have been staged and notification has been received by the CALDB manager, the CALDB manager will:

1. Copy the data files to the appropriate subdirectory of the HEASARC (public) CALDB
2. Verify the data files which have been copied using the `CHECKSUM/DATASUM` keywords
3. Send a confirmation e-mail to the Project Calibration Team that the data have been copied. The body of this e-mail will consist of a list of the copied files in the same format as used in the notification e-mail. The subject line of this e-mail will include the name of the CALDB update, as in:

   ```
   Subject: caldb update ASCA19931220 files copied
   ```

4. The privileges on the copied files in the HEASARC CALDB will be changed to allow world read access.